软件测试上机报告



第六次上机作业 – MuJava

**学 院 智能与计算学部**

**专    业 软件工程**

**姓 名 白文杰**

**学 号 3018216031**

**年 级 2018**

**班    级 1**

#### Experimental requirements

1. Install MuJava. The instruction of how to install and use Mujava can be seen in <https://cs.gmu.edu/~offutt/mujava/> .
2. Two small programs are given for your task. BubbleSort.java is an implementation of bubble sort algorithm and BackPack.java is a solution of 01 backpack problem. Try to generate Mutants of 2 given programs with MuJava.
3. Write testing sets for 2 programs with Junit, and run mutants on the test sets with MuJava.

**Requirements for the experiment:**

1. Finish the tasks above individually.
2. Check in your java code to github or gitee.
3. Post your experiment report to “智慧树” , the following information should be included in your report:
   1. The brief description that you install MuJava
   2. Steps for generating Mutants
   3. Steps for making test sets and running mutants.
   4. Your mutants result (The number of live mutants, killed mutants, etc.)

#### Configuration

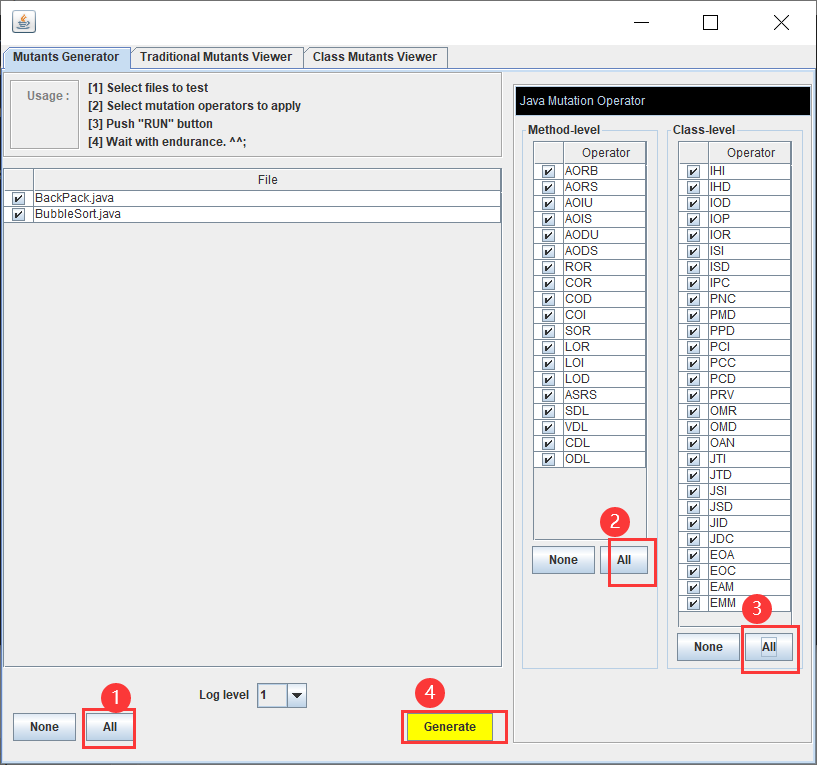
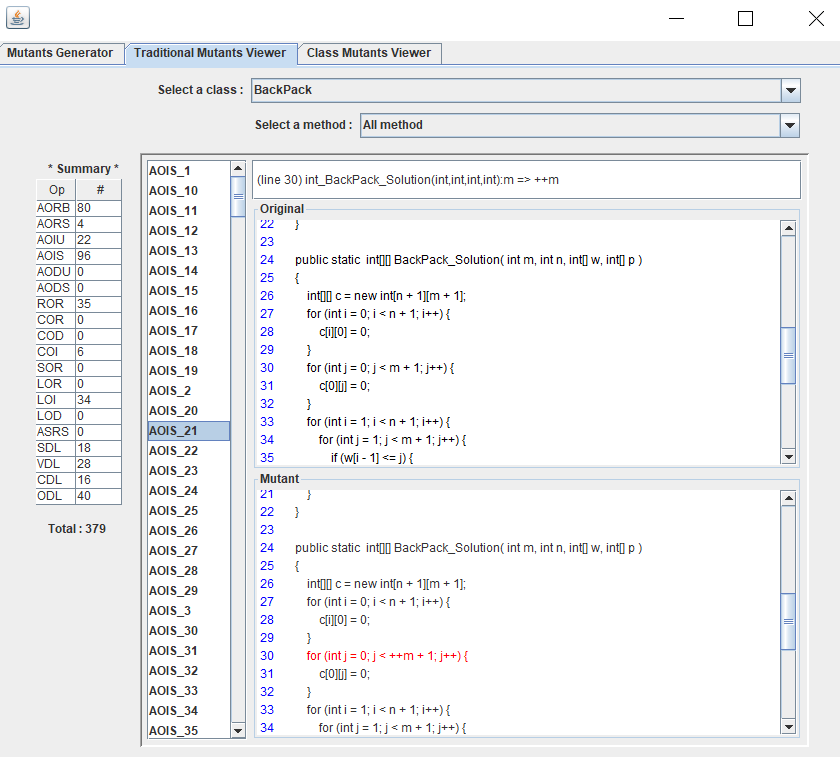
* + - 1. 首先下载openjava.jar和mujava.jar，以及自己手写一个mujava.config指向存放mujava的目录，并将这些文件放在mujava目录下。此时，基本的mujava环境就搭建好了。
      2. 分别创建src（存放源码文件）、classes（存放源码文件的编译结果）、result（存放变异结果）、testset（Junit测试文件和测试文件编译结果）
      3. 创建两个脚本，分别命名为GenMutants.bat（生成变异结果）、RunTest.bat（对变异结果进行测试）。两个文件的源码见Part 4 Source Code。
      4. 首先执行GenMutants.bat，此时会弹出一个GUI界面，如下：  
           
         它会自动查找classes目录下的class文件，然后依次按照图示顺序点击按钮即可生成变异体（存放在result目录下）。此时可以点击Traditional Mutants Viewer和Class Mutants Viewer可以查看传统变异和类变异的结果，如下图：  
         

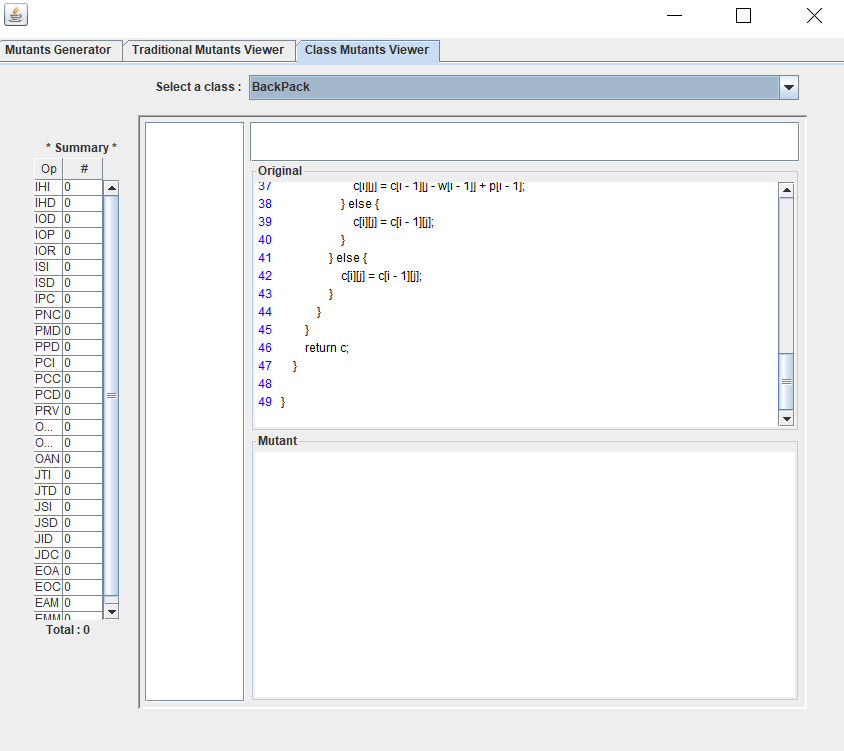
图 1 BackPack传统变异结果

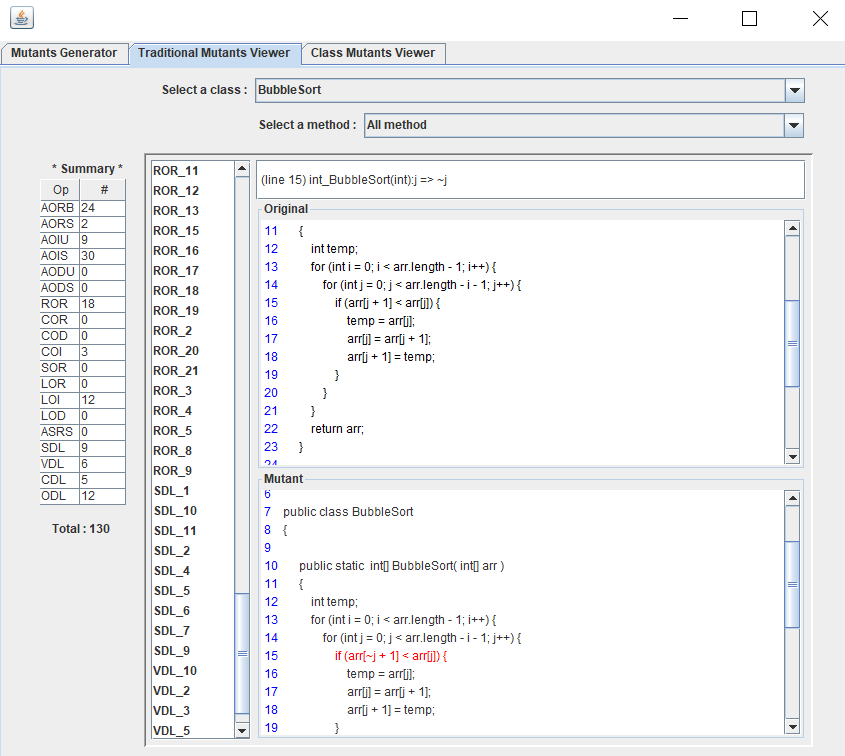
图 2 BackPack类变异结果

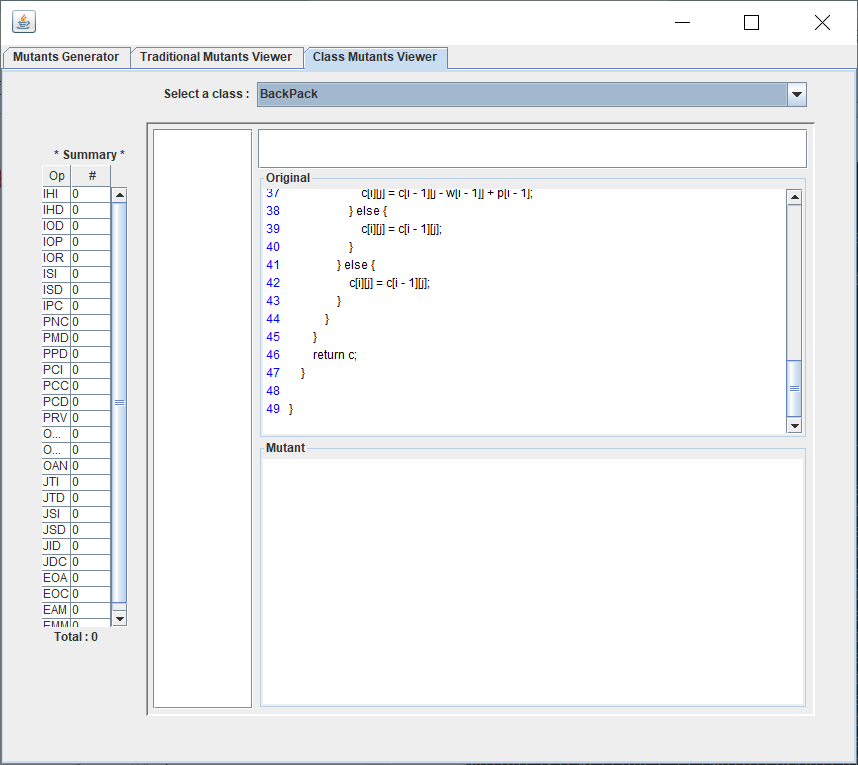
图 3 BubbleSort传统变异结果

图 4 BubbleSort类变异结果

1. 执行RunTest.bat，此时也会有一个GUI界面，其作用是测试之前编译的结果是否能够满足测试用例。选择要测试的类和其对应的测试类，点击Run按钮即可。如下图：

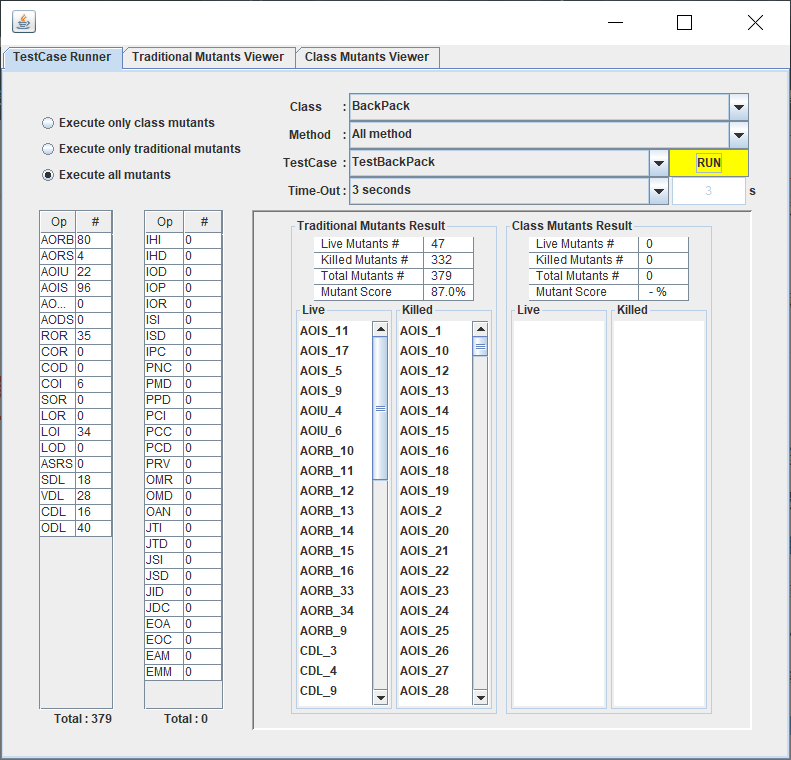


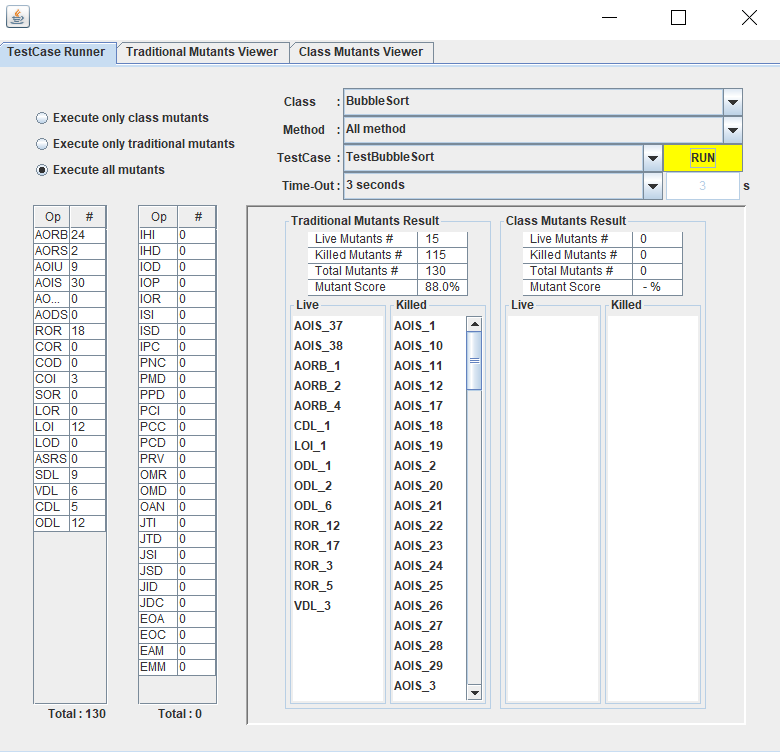
图 5 BackPack杀死率  


图 6 BubbleSort杀死率

#### Result analysis

从两个程序的杀死率来看，变异结果的88%都是错误的代码，不能够满足我们正确的逻辑。

在执行测试的时候，我遇到了杀死率为0的问题。出现问题的主要原因是：我把当前目录添加到了CLASSPATH环境变量中，而当前目录包含classes目录，这样导致变异体被“绕过”了，直接测试了源文件，所以全部执行通过了。

#### Source code

* + - 1. GenMutants.bat

|  |
| --- |
| **set** CLASSPATH**=%CLASSPATH%**;.;C:\MuJava\mujava.jar;C:\MuJava\openjava.jar;D:\Java\jdk1.8.0\_102\lib\tools.jar;  **cd** C:\MuJava  java mujava.gui.GenMutantsMain |

* + - 1. RunTest.bat

|  |
| --- |
| **set** CLASSPATH**=**C:\MuJava\mujava.jar;C:\MuJava\openjava.jar;D:\Java\jdk1.8.0\_102\lib\tools.jar;C:\MuJava\junit-4.13.1.jar;C:\MuJava\hamcrest-all-1.3.jar;  **cd** C:\MuJava  java mujava.gui.RunTestMain **>** output.txt |